A New Approach for the Bearings-Only Problem: estimation of the variance-to-range ratio

Thomas Bréhard
IRISA/CNRS
IRISA, Campus de Beaulieu
35042 Rennes Cedex
France
tbrehard@irisa.fr

Jean-Pierre Le Cadre
IRISA/CNRS
IRISA, Campus de Beaulieu
35042 Rennes Cedex
France
lecadre@irisa.fr

Abstract – *The classical bearings-only tracking problem (BOT) for a single object belongs to the class of non linear filtering problems. Recently, algorithms based on sequential Monte Carlo methods (particle filtering) have been proposed in the modified polar coordinate (MP) framework. This latter has been shown to be fundamentally relevant in this context as regards observability and initialization problems. In this paper, we here address a more general class of problems: the non linear filtering problems with unknown variance state. In this context, only a weak prior information is assumed on the temporal evolution of the target which is an important issue in practice. Our original contribution is an algorithm which is able to estimate the* **variance-to-range ratio**. *As a by product, it is shown that this ratio is always observable.*

Keywords: bearings-only tracking, covariance estimation, initialization, sequential Monte Carlo methods, modified polar coordinates.

1 Introduction

The aim of BOT is to determine the trajectory of a target using noisy bearing measurements from a single observer. Let us assume that the target motion may be described by a diffusion model (see [1] for an exhaustive review on dynamic models). The problem is classically composed of two stochastic equations. The first one represents the temporal evolution of the target state (position and velocity) and is called state equation. The second one links the bearing measurement to the state of the target at time t (measurement equation). Non-linearity of the measurement equation is a main difficulty. Particle filtering [2, 3, 4] is now the method of reference.

An original extension of the BOT problem named "σ-BOT" is studied where the state covariance σ which represents the maneuverability of the target is unknown. This is an important issue in practice. We generally do not know if the target goes straight line (σ is null) or maneuvers. Consequently this parameter is unknown and must be learned. However the problem belongs to the class of non linear filtering problems with unknown variance state. Furthermore Mehra in [5] has shown in the linear case that the process noise covariance can be estimated using the innovation sequence. Then the question is can we use this idea in the "σ-BOT" context ?

First of all, the σ-BOT problem is presented in section 2 using the cartesian coordinate framework. Otherwise, the modified polar (MP) coordinate system introduced by Aidala and Hammel in [6] is fundamentally relevant in the classical BOT context in particular for the initialization of the particle filter [7] and for deriving a closed-form solution in the deterministic case [8]. We deduce from this framework that $\frac{\sigma}{r(t)}$ named "variance-to-range ratio" is the natural process noise covariance that can be estimated using the innovation sequence even if the range itself is not observable (i.e. the observer is not maneuvering). More generally, it appears that one more time, the MP coordinate system is relevant in the "σ-BOT" context. Consequently, our original contribution is a particle filtering algorithm, especially designed for the "σ-BOT" context.

2 The $\sigma-$BOT problem

2.1 The $\sigma-$BOT in cartesian coordinate system

Historically, BOT is presented in the cartesian coordinate system. Let us define:

$$X_t = \begin{pmatrix} X_1(t) \\ X_2(t) \\ X_3(t) \\ X_4(t) \end{pmatrix} = \begin{pmatrix} v_x(t) \\ v_y(t) \\ r_x(t) \\ r_y(t) \end{pmatrix} \text{ and } \sigma, \tag{1}$$

the state of the target at time t composed of relative velocity and position of the target in the $x-y$ plane and the state variance named "σ". This latter quantity represents the maneuverability of the target. It is assumed that the target follows a nearly constant-velocity model. The discretized state equation [1] is then:

$$X_{t+1} = FX_t + HU_t + \sigma W_t \ , \tag{2}$$

where:

$$\left| \begin{array}{l} W_t \sim \mathcal{N}(0, Q) \ , \\ F = \begin{pmatrix} 1 & 0 \\ \delta_t & 1 \end{pmatrix} \otimes Id_2 \ , \\ H = \begin{pmatrix} 1 \\ \delta_t \end{pmatrix} \otimes Id_2 \ , \\ Q = \begin{pmatrix} \delta_t & \frac{\delta_t^2}{2} \\ \frac{\delta_t^2}{2} & \frac{\delta_t^3}{3} \end{pmatrix} \otimes Id_2 \ , \end{array} \right. \tag{3}$$

[1]For a general review of dynamic models for target tracking see [1].

δ_t is the elementary time period and U_t is the known difference between observer velocity at time $t+1$ and t (observer maneuvers).

Otherwise, we note Z_t the bearing measurement received at time t. The target state is related to this measurement through the following equation:

$$Z_t = \tan^{-1}\left(\frac{r_x(t)}{r_y(t)}\right) + V_t \ , \qquad (4)$$

where $V_t \sim \mathcal{N}(0, \sigma_v^2)$. The measurement variance σ_v^2 is known. The system (2–4) has two components : a linear state equation (2) and a non linear measurement equation (4). Particle filter techniques (see [2, 3, 4]) are, thus, quite relevant.

2.2 The $\sigma-$BOT in MP coordinate system

However, as shown in [9] a problem of observability is hidden in the cartesian formulation. As a matter of fact, no information on range exists *as long as* the observer is not maneuvering. So, the idea consists in using a coordinate system for which the unobservable component (range) is not coupled with the observable components. This is the motivation of Aidala and Hammel [6] for defining MP system. We add a fifth component, namely the **variance-to-range ratio** ρ_t to the classical MP components. We denote:

$$Y_t = \begin{pmatrix} Y_1(t) \\ Y_2(t) \\ Y_3(t) \\ Y_4(t) \end{pmatrix} = \begin{pmatrix} \dot\beta(t) \\ \frac{\dot r(t)}{r(t)} \\ \beta(t) \\ \frac{1}{r(t)} \end{pmatrix} \text{ and } \rho_t = \frac{\sigma}{r(t)} \ , \qquad (5)$$

the target state at time t in MP coordinate system where $\beta(t)$ and $r(t)$ are the relative bearing and the target range, respectively. $\dot\beta(t)$ and $\dot r(t)$ are the time derivative of $\beta(t)$ and $r(t)$. The aim of this section consists in showing that this is the most natural parameterization of the $\sigma-$BOT in MP coordinate system.

First let us remark that the stochastic system (2–4) becomes:

$$\begin{aligned} Y_{t+1} &= f_c^{mp}\left[F f_{mp}^c(Y_t) + HU_t + \sigma W_t\right], \qquad (6) \\ Z_t &= H(f_{mp}^c(Y_t)) + V_t, \qquad (7) \end{aligned}$$

where f_c^{mp} and f_{mp}^c are cartesian-to-MP and MP-to-cartesian state mapping functions such that:

$$\begin{aligned} X(t) &= f_{mp}^c(Y(t)) \qquad (8) \\ &= \frac{1}{Y_4(t)} \begin{pmatrix} Y_2(t)\sin(Y_3(t)) + Y_1(t)\cos(Y_3(t)) \\ Y_2(t)\cos(Y_3(t)) - Y_1(t)\sin(Y_3(t)) \\ \sin(Y_3(t)) \\ \cos(Y_3(t)) \end{pmatrix}, \end{aligned}$$

and

$$\begin{aligned} Y(t) &= f_c^{mp}(X(t)) \qquad (9) \\ &= \begin{pmatrix} \frac{X_1(t)X_4(t) - X_2(t)X_3(t)}{X_3^2(t)+X_4^2(t)} \\ \frac{X_1(t)X_3(t)+X_2(t)X_4(t)}{X_3^2(t)+X_4^2(t)} \\ \tan^{-1}\left(\frac{X_3(t)}{X_4(t)}\right) \\ \frac{1}{\sqrt{X_3^2(t)+X_4^2(t)}} \end{pmatrix}. \end{aligned}$$

Now we are going to pay more attention to the stochastic system (6–9) using Aidala and Hammel's formulation of the problem. Expliciting f_c^{mp} and f_{mp}^c, Eqs.(6–9) can be rewritten:

$$\begin{pmatrix} Y_1(t+1) \\ Y_2(t+1) \\ Y_3(t+1) \end{pmatrix} = \begin{pmatrix} \frac{S_1(t)S_4(t)-S_2(t)S_3(t)}{S_3^2(t)+S_4^2(t)} \\ \frac{S_1(t)S_3(t)+S_2(t)S_4(t)}{S_3^2(t)+S_4^2(t)} \\ Y_3(t) + \tan^{-1}\left(\frac{S_3(t)}{S_4(t)}\right) \end{pmatrix} \qquad (10)$$

$$Y_4(t+1) = \frac{Y_4(t)}{\sqrt{S_3^2(t)+S_4^2(t)}} \ , \qquad (11)$$

$$\rho_{t+1} = \frac{\rho_t}{\sqrt{S_3^2(t)+S_4^2(t)}} \ , \qquad (12)$$

$$Z_t = Y_3(t) + V_t \ , \qquad (13)$$

where:

$$\begin{aligned} \begin{pmatrix} S_1(t) \\ S_2(t) \\ S_3(t) \\ S_4(t) \end{pmatrix} &= \begin{pmatrix} Y_1(t) \\ Y_2(t) \\ \delta_t Y_1(t) \\ 1 + \delta_t Y_2(t) \end{pmatrix} \qquad (14) \\ &+ Y_4(t)\left[\begin{pmatrix} 1 \\ \delta_t \end{pmatrix} \otimes P_{Y_3(t)} U_t\right] + \rho_t W_t \ , \end{aligned}$$

and

$$P_{Y_3(t)} = \begin{pmatrix} \cos(Y_3(t)) & -\sin(Y_3(t)) \\ \sin(Y_3(t)) & \cos(Y_3(t)) \end{pmatrix} . \qquad (15)$$

Eqs. (10,11) are proved in [6]. Eq. (12) is easily obtain using Eq. (11) and the definition of the state variance-to-range ratio. Finally, if we note:

$$Y_t^r = \begin{pmatrix} Y_1(t) \\ Y_2(t) \\ Y_3(t) \end{pmatrix} = \begin{pmatrix} \dot\beta(t) \\ \frac{\dot r(t)}{r(t)} \\ \beta(t) \end{pmatrix} , \qquad (16)$$

then we can write Eqs. (10–15) according to the last notations:

$$\begin{aligned} Y_{t+1}^r &= F_1(\ Y_t^r \ , \ Y_4(t)U_t \ , \ \rho_t\ W_t\ )\ , \qquad (17) \\ Y_4(t+1) &= Y_4(t)F_2(\ Y_t^r,\ \rho_t\ W_t\ )\ , \qquad (18) \\ \rho_{t+1} &= \rho_t F_2(\ Y_t^r,\ \rho_t\ W_t\ )\ , \qquad (19) \\ Z_t &= Y_3(t) + V_t\ . \qquad (20) \end{aligned}$$

As in the cartesian framework, the stochastic system (17–20) is a non linear filtering problem with unknown state covariance. The only difference is that the state covariance ρ_t depends on time in the MP formulation. Moreover one can see that in the case of a non-maneuvering observer (U_t is a zero vector), $Y_4(t)$ is unobservable because it does not appear in Eqs. (17,19,20). However ρ_t, for its own, is (stochastically) observable as it will be shown in the next section.

3 Stochastic observability of the variance-to-range ratio

The aim of this section is to give the intuition that the variance-to-range ratio is observable. A simple way consists in producing a simple estimator. We restrict here to the case where the observer is not maneuvering i.e. U_t is null. Let us first consider a second order expansion of Eq. (17):

$$Y_{t+1}^r = \tilde{F}_1(Y_t^r) + \rho_t \; \mathcal{W}_t(Y_t^r) \; , \tag{21}$$

where $\mathcal{W}_t(Y_t^r) \sim \mathcal{N}(0, \tilde{Q}_{Y_t^r})$. Let us notice that the covariance matrix $\tilde{Q}_{Y_t^r}$ depends on the observable components Y_t^r. Now, if we note :

$$\epsilon_t^2 = (Y_{t+1}^r - \tilde{F}_1(Y_t^r))^H \tilde{Q}_{Y_t^r}^{-1} (Y_{t+1}^r - \tilde{F}_1(Y_t^r)) \; , \tag{22}$$

we have

$$\frac{\epsilon_t^2}{\rho_t^2} \sim \mathcal{X}^2(3) \; . \tag{23}$$

Thus we can see that ρ_t^2 is the covariance of ϵ_t. Consequently, ρ_t can be estimated if we have an estimate of ϵ_t's law. Now let $\hat{\rho}_t$ be our estimator such that:

$$\hat{\rho}_t = \arg\max p(\rho_t | Z_0, \ldots, Z_t). \tag{24}$$

Then

$$\hat{\rho}_t = \arg\max \int p(\rho_t|\epsilon_t) p(\epsilon_t | Z_0, \ldots, Z_t) d\epsilon_t. \tag{25}$$

Otherwise Y_t^r is observable at each step of time so we can have an estimate of $p(\epsilon_t|Z_0, \ldots, Z_t)$ denoted $\hat{p}(\epsilon_t|Z_0, \ldots, Z_t)$ such that:

$$\hat{\rho}_t = \arg\max \int p(\rho_t|\epsilon_t) \hat{p}(\epsilon_t | Z_0, \ldots, Z_t) d\epsilon_t. \tag{26}$$

Consequently, an estimate can be computed using Monte Carlo simulations and based on Eqs. (23,26) . Let us remark that ρ_t is a covariance term also one can expect a low convergence speed. Now a definitive approach may be to compute the posterior Cramér-Rao bound for the variance-to-range ratio. Otherwise this estimator will not be the preferred way in practice, essentially because it is based on a linearization. Another estimate of the variance-to-range ratio can be obtained directly form the particle filtering algorithm developed in section 4.

Finally we can now define an "estimability" order for the target state estimation:

1. $Y_3(t)$ is the more estimable component because it is obtained directly from the measurement.
2. Then comes $Y_1(t)$ which is just the time derivative of $Y_3(t)$.
3. Next, $Y_2(t)$ which is a function of the first and second time derivative of $Y_3(t)$.
4. Then ρ_t which can be computed using the noise variance of the Markov process relying Y_{t+1}^r to Y_t^r.
5. Finally, $Y_4(t)$ is only observable when the observer is maneuvering.

3.1 An extension of the σ-BOT

We assume now that the standard deviation of state equation may be described by a diffusion model such that:

$$\begin{aligned} X_{t+1} &= AX_t + HU_t + \sigma_t W_t \; , & (27) \\ \sigma_{t+1} &= G(\sigma_t, \eta_t) \; , & (28) \\ Z_t &= \tan^{-1}\left(\frac{r_x(t)}{r_y(t)}\right) + V_t \; , & (29) \end{aligned}$$

where G is a possible non linear function and η_t a noise process. This formulation can be quite relevant in the detection of maneuvers context. Then this problem can be wrote using the MP framework:

$$\begin{aligned} Y_{t+1}^r &= F_1(\; Y_t^r \; , \; Y_4(t)U_t \; , \; \rho_t \; W_t \;) \; , & (30) \\ Y_4(t+1) &= Y_4(t) F_2(\; Y_t^r , \; \rho_t \; W_t \;) \; , & (31) \\ \rho_{t+1} &= \rho_t F_2(\; Y_t^r , \; \rho_t \; W_t \;) \frac{G(\sigma_t, \eta_t)}{\sigma_t} \; , & (32) \\ Z_t &= Y_3(t) + V_t \; . & (33) \end{aligned}$$

The σ-BOT problem is of course a particular case where $\frac{G(\sigma_t,\eta_t)}{\sigma_t}$ is equal to 1. Furthermore ρ_t is still observable in this general problem. This holds for σ_t too if $G(\sigma_t, \eta_t)$ is not proportional to σ_t. It could be particularly interesting to investigate the performance analysis using the PCRB.

4 Particle filtering algorithm for the σ-BOT problem

Particle filtering algorithms are generally composed of three stages at each step of time. First, a particle set representing different possible states of the target is propagated using the state equation. Second, the weights of the particles are updated according Bayes's formula using the measurement equation. The state distribution is a finite weighted sum of Dirac laws centered around the particles. The third stage is a resampling step in order to avoid degeneracy of the particle set. It may be mentioned in passing that many ways have been developed to improve particle filtering algorithms: the use of kernel filter has been studied in [10] as well as the resampling frequency in [11].

The aim of this section is the initialization of particle filtering which is one of the main difficulty, as well as the estimation and the resampling steps of the particle filtering algorithm which are not classical.

4.1 Initialization of the particle filtering algorithm

This method proposed in [7] consists in determining the batch duration sufficient for ensuring a good initialization of the particle filtering algorithm. The three first components of the particles are then initialized by sampling uniformly in a confidence area. Moreover, the fourth component and ρ_t are sampling uniformly using a weak prior information.

4.1.1 Initialization of the set of particles

Assuming that the target motion is deterministic, the stochastic system (17–19) becomes:

$$Z_t = Y_3(k) + \tan^{-1}\left(\frac{(t-k)\delta_t Y_1(k)}{1+(t-k)\delta_t Y_2(k)}\right) + V_t \; , \quad \forall t \geq 0 \; , \tag{34}$$

which is a non linear regression problem. Let us denote $\hat{Y}_k^r$, the maximum likelihood estimator (MLE) of the observable components of the state at time k using the $2k+1$ first bearing measurements. It is computed by means of a Gauss-Newton algorithm since eq.(34) is non-linear. Moreover, using classical convergence results, we can define a confidence area noted $CA(\hat{Y}_k^r)$ for the MLE. Then the three first components of the particles can be initialized by sampling uniformly in $CA(\hat{Y}_k^r)$ such that:

$$CA(\hat{Y}_k^r) = \left\{ Y_k^r \;\middle|\; \left\| \hat{Y}_k^r - Y_k^r \right\|^2_{J(\hat{Y}_k^r)^{-1}} \leq \frac{\mathcal{X}_3^2(1-\alpha)}{2k+1} \right\} \; , \tag{35}$$

where $J(Y_k^r)$ is the Fisher information matrix. It is worth stressing that $CA(\hat{Y}_k^r)$ is an hyperellipsoid. Then the initialization of the observable components of the state of the particles can be done using the algorithm proposed by Dezert and Musso in [12].

It remains finally to fix $Y_4(k)$ and $Y_5(k)$ the fourth and the fifth component of each of the particles. Let us remark that $Y_4(k)$ is the inverse of the range at time k. If we assume that:

$$r_{min} \leq r(t) \leq r_{max} \; , \tag{36}$$

then an intuitive idea consists in giving to each particle a range value uniformly sampled between a minimum and a maximum relative target range noted R_{min} and R_{max}. Moreover, if we assume that:

$$\sigma_{min} \leq \sigma \leq \sigma_{max} \; , \tag{37}$$

then the fifth component can be uniformly sampled between $\frac{\sigma_{min}}{R_{max}}$ and $\frac{\sigma_{max}}{R_{min}}$.

4.1.2 Estimation of the batch duration

It remains now to determine the batch duration, sufficient for ensuring a good initialization of the particle filtering algorithm. Intuitively, the volume of $CA(\hat{Y}_k^r)$ decreases with the time k. If we associate to each of the particles a neighborhood such that the true state of the target is lying in (at least) one of these neighborhoods, then the problem of the choice of k reverts to determining the batch duration which ensures that N particles are sufficient to fill the confidence area.

For a given particle (i), this neighborhood represents the capacity of the particle filter to tend toward the true state. This latter can be defined using a linearization of the diffusion model. Let us denote $\mathcal{V}(\mathcal{B}(Y_k^{(i)}))$ the volume of the neighborhood of the particle (i) in MP coordinate system. Moreover we define the confidence area for $\hat{Y}_k$ using both $CA(\hat{Y}_k^r)$ and the prior information relative to $Y_4(k)$. Practically, this means that the particle filter can be initialized as soon as the following condition holds:

$$\mathcal{V}(CA(\hat{Y}_k))) \leq \sum_{i=1}^{N} \mathcal{V}(\mathcal{B}(Y_k^{(i)})) \; . \tag{38}$$

One can show that

$$\mathcal{V}(\mathcal{B}(Y_k^{(i)})) \leq \frac{(\pi \mathcal{X}_4^2(1-\alpha))^2 (Y_4^{(i)}(k-1))^5 \sqrt{det(\sigma_{max}^2 Q)}}{\Gamma(3)} \; , \tag{39}$$

where $\Gamma(.)$ is the classical gamma function and $det(.)$ the determinant function. Finally,

$$\mathcal{V}(CA(\hat{Y}_k)) \approx \left(\frac{1}{R_{min}} - \frac{1}{R_{max}}\right) \frac{(\pi \mathcal{X}_4^2(1-\alpha))^{3/2}}{\Gamma(5/2)\sqrt{det(J(\hat{Y}_k^r))}} \; . \tag{40}$$

The initialization method is sum up in Fig.1.

- k=3
- While $\mathcal{V}(CA(\hat{Y}_k)) > \sum_{i=1}^{N} \mathcal{V}(\mathcal{B}(Y_k^{(i)}))$
  1. Estimate $\hat{Y}_k^r$ using a Gauss-Newton iterative algorithm.
  2. Compute $\mathcal{V}(CA(\hat{Y}_k))$ using Eq.(40).
  3. Compute $\mathcal{V}(\mathcal{B}(Y_k^{(i)}))$ for $i = 1, \ldots, N$ using Eq.(39).
  4. $k = k + 1$.
- Initialization of the particles, for $i = 1, \ldots, N$

$$\begin{pmatrix} Y_1^{(i)}(k) \\ Y_2^{(i)}(k) \\ Y_3^{(i)}(k) \end{pmatrix} \sim \mathcal{U}(CA(\hat{Y}_k^r)) \; , \tag{41}$$

$$Y_4^{(i)}(k) \sim \frac{1}{\mathcal{U}([R_{min}, R_{max}])} \; , \tag{42}$$

$$\rho_k^{(i)} \sim \mathcal{U}\left(\left[\frac{\sigma_{min}}{R_{max}}, \frac{\sigma_{max}}{R_{min}}\right]\right) \; . \tag{43}$$

Fig. 1: Initialization of particle filtering algorithm in MP coordinates.

4.2 Estimation and resampling in the particle filtering algorithm

The important point is that as long as the observer is not maneuvering, the fourth component of the state is not coupled with the other components of the state. Consequently until the observer maneuvers, we estimate the target state such as:

$$\hat{\mathbb{E}}(Y_k(t)) = \sum_{i=1}^{N} q_t^{(i)} Y_k^{(i)}(t) \,, \quad \text{for } k = \{1,2,3\}, \tag{44}$$

$$\hat{\mathbb{E}}(Y_4(t)) = \frac{1}{N}\sum_{i=1}^{N} Y_4^{(i)}(t) \,, \tag{45}$$

$$\hat{\mathbb{E}}(\rho_t) = \sum_{i=1}^{N} q_t^{(i)} \rho_t^{(i)} \,, \tag{46}$$

where $\{q_t(1), \ldots, q_t(N)\}$ is the set of normalized weights obtained by particle filtering. One can remark that the estimate of $Y_4(t)$ is the same at each step of time.

Otherwise, all the components of the state except the fourth component are resampled to ensure the independence property between $Y_4(t)$ and the other components in the resampling step. For $i = 1, \ldots, N$:

$$Y_k^{(i)}(t) \sim \sum_{j=1}^{N} q_t^{(j)} \mathbb{1}_{Y_k^{(j)}(t)}, \quad \text{for } k = \{1,2,3\}, \tag{47}$$

$$\rho_t^{(i)} \sim \sum_{j=1}^{N} q_t^{(j)} \mathbb{1}_{\rho_t^{(j)}} \,. \tag{48}$$

Finally, as soon as the observer is maneuvering, the fourth component is now coupled to the other components of the state. Then, we use the classical method to estimate and resample $Y_4(t)$:

$$\hat{\mathbb{E}}(Y_4(t)) = \sum_{i=1}^{N} q_t^{(i)} Y_4^{(i)}(t), \tag{49}$$

$$Y_4^{(i)}(t) \sim \sum_{j=1}^{N} q_t^{(j)} \mathbb{1}_{Y_4^{(j)}(t)}. \tag{50}$$

The particle filtering algorithm is sum up in Fig. 2. It must be noticed here that the particle filtering algorithm must use the modified polar coordinate system before the observer maneuvers.

5 Simulation results

Let us now illustrate the performance of particle filtering algorithm described in Fig. 2. This latter has been programmed in Matlab. The parameters involved in the algorithm are put together in Tab.1.

Two different scenarios have been studied. As for the computation cost, Tab. 2 contains the cost for one iteration of the initialization algorithm and the time spend on a particle filtering algorithm iteration on a 2.6 $Ghz\ Pentium\ IV$.

- Initialization (see Fig. 1),
- While observer is not maneuvering ($U_t = 0$):
  1. Diffusion of the particles using Eqs. (10-12)
  2. Weighting: for $i = 1, \ldots, N$
     $$q_t^{(i)} \propto q_{t-1}^{(i)} e^{\frac{(Z_t - Y_3(t)^{(i)})^2}{2\sigma_w^2}} \tag{51}$$
  3. Estimation using Eqs. (44–46)
  4. if $\frac{1}{\sum_{i=1}^{N} q_t^{(i)}} < N_{threshold}$
     resample using Eqs. (47,48)
  5. t=t+1
- When observer has already maneuvered:
  1. Diffusion of the particles using Eqs. (10–12)
  2. Weight the particles using Eq. (51)
  3. Estimation using Eqs. (44,49,46)
  4. if $\frac{1}{\sum_{i=1}^{N} q_t^{(i)}} < N_{threshold}$
     resample using Eqs. (47,50,48)
  5. t=t+1

Fig. 2: Particle filtering algorithm in MP coordinates.

5.1 First scenario

The following scenario is considered. The initial states of the observer and the target are:

$$X_0^{obs} = \begin{pmatrix} -10 \text{ ms}^{-1} \\ 2 \text{ ms}^{-1} \\ 10000 \text{ m} \\ 0 \text{ m} \end{pmatrix}, \tag{52}$$

$$X_0^{target} = \begin{pmatrix} 8 \text{ ms}^{-1} \\ -3 \text{ ms}^{-1} \\ -5000 \text{ m} \\ 10000 \text{ m} \end{pmatrix}. \tag{53}$$

The relative target state at initial time is then given by $X_0 = X_0^{target} - X_0^{obs}$. The elementary time period δ_t is 6 s. The standard deviation of the process noise in the state equation σ is fixed to $0.04\,\text{ms}^{-1}$ so that target trajectory strongly departs from a straight line. The standard deviation of the measurement noise σ_w is 0.05 rad (about 3 deg.). An example of trajectory is presented in Fig. 3(a), while a bearing measurement batch is presented in Fig. 3(b).

In Fig. 5 simulation results are presented. At the beginning of the scenario, the estimated components are restricted to the observable ones i.e. $\{Y_1(t), Y_2(t), Y_3(t)\}$ as solution of the non-linear regression problem Eq. (34). Of course, we do not have an estimate for ρ_t. At time 1128, the "initialization condition" Eq. (38) turns to be true which means that we are able to initialize the particle filtering algorithm at time 564. From this time, the tracking algorithm estimates the full state of the target. The first (three) components of target state are correctly estimated form the beginning thanks to the initialization method. Finally we can see in (Fig.5, d) that the confidence area related to the variance-to-range ratio ρ_t is very high at the beginning but decreases over the time. This component of the state is accurately estimated at time 2000.

In this scenario, target state was not difficult to estimate because the bearing variations were high as we can see in Fig. 3(b). Now let us study a more difficult scenario.

5.2 Second scenario

In the second scenario, the observer follows the same trajectory as in the first one. The initial state of the target is:

$$X_0^{target} = \begin{pmatrix} -8 \text{ ms}^{-1} \\ 3 \text{ ms}^{-1} \\ -5000 \text{ m} \\ 10000 \text{ m} \end{pmatrix} . \qquad (54)$$

An example of trajectory is presented in Fig. 4(a), while a bearing measurement batch is presented in Fig. 4(b). We can see in this case that the bearing variations were smaller than in the first scenario. The simulation results are presented in Fig. 6. Of course the third component β_t is correctly estimated as well as the first component $\dot{\beta}_t$. Otherwise, we can notice that bearing variations are too weak for estimating the second component $\frac{\dot{r}_t}{r_t}$ and the variance-to-range ratio ρ. For this scenario, useful information relative to ρ is an upper bound given by the 2 σ confidence interval; showing moreover that this is a positive variable.

6 Conclusion

An extension of the BOT problem named σ-BOT have been studied here. In this case, variance in state equation is assumed unknown which his is an important issue in practice. An original parameterization has been defined for the problem composed of the classical polar modified coordinate system and the variance-to-range ratio. We have shown that this ratio is observable without information on range. Then a solution to the σ-BOT have been proposed based on particle filtering techniques. The algorithm performs quite satisfactorily. Future developments include the study of the variance-to-range ratio specially the posterior Cramér-Rao bound as well as the use of this parameter in multi-target environment or target classification problem.

References

[1] X. Li and V. Jilkov. A Survey of Manoeuvring Target Tracking: Dynamics Models. In *SPIE Conf. on Signal and Data Processing of Small Targets*, Florida, USA, April 2000.

[2] N. Gordon, D. Salmond, and A. Smith. Novel Approach to Non-Linear/Non-Gaussian Bayesian State Estimation. *Proc. Inst. Elect. Eng. F*, 140(2):107–113, 1993.

[3] A. Doucet, N. De Freitas, and N. Gordon. *Sequential Monte Carlo Methods in Practice*. Springer-Verlag, New-York, 2001.

[4] M.S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A Tutorial on Particule Filters for Online Non-Linear/Non-Gaussian Bayesian Tracking. *IEEE Trans. Signal Processing*, 50(2):174–188, 2002.

[5] R. K. Mehra. On the identification of variances and adaptive kalman filtering. *IEEE Trans. on Automat. Control.* , 15:175–184, April 1970.

[6] V.J. Aidala and S.E. Hammel. Utilization of Modified Polar Coordinates for Bearing-Only Tracking. *IEEE Trans. Automatic Control*, 28(3):283–294, March 1983.

[7] T. Bréhard and J-P. Le Cadre. Initialization of particle filter and posterior cramér-rao bound for bearings- only tracking in modified polar coordinate system. *Submitted for publication in IEEE Transactions on Aerospace., Electronic and Systems*, 2004.

[8] S.C. Nardone and M.L. Graham. A Closed-form Solution to Bearing-Only Target Motion Analysis. *IEEE Journal of Oceanic Engineering*, 22(1), january 1997.

[9] S.C. Nardone and V.J. Aidala. Observability Criteria for Bearings-0nly Target Motion Analysis. *IEEE Trans. on Aerospace, Electronic and Systems*, 17:161–166, march 1981.

[10] N. Oudjane C. Musso and F. Legland. Improving regularised particle filters. In A. Doucet, N. De Freitas, and N. Gordon, editors, *Sequential Monte Carlo Methods in Practice*, New-York, 2001. Springer-Verlag.

[11] C. Hue, J-P. Le Cadre, and P. Pérez. Sequential Monte Carlo Methods for Multiple Target Tracking and Data Fusion. *IEEE Trans. on Signal Processing*, 50(2):309–325, February 2002.

[12] J. Dezert and C. Musso. An Efficient Method for Generating Points Uniformly Distributed in Hyperellipsoids. In *Proceedings of the Workshop on Estimation, Tracking and Fusion: A Tribute to Yaakov Bar-Shalom*, may 2001.

parameter	**value**
R_{min}	5000m
R_{max}	20000m
σ_{min}	0.01 ms^{-1}
σ_{max}	0.05 ms^{-1}
N	5000
$N_{threshold}$	0.9

Table 1: Parameters for the particle filtering algorithm

Iteration	**Cost**
initialization	about 70 ms
particle filtering algorithm	about 250 ms

Table 2: Computation cost for one iteration on a 2.6 $Ghz\ Pentium\ IV$

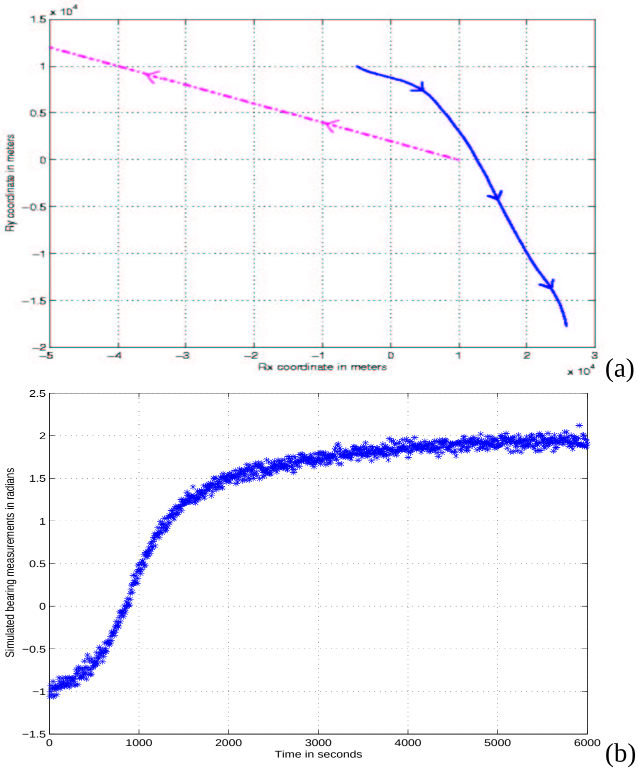

Fig. 3: *Scenario 1*: (a) trajectories of the observer (red dashed line) and the target (blue solid line). (b) Simulated bearing measurements

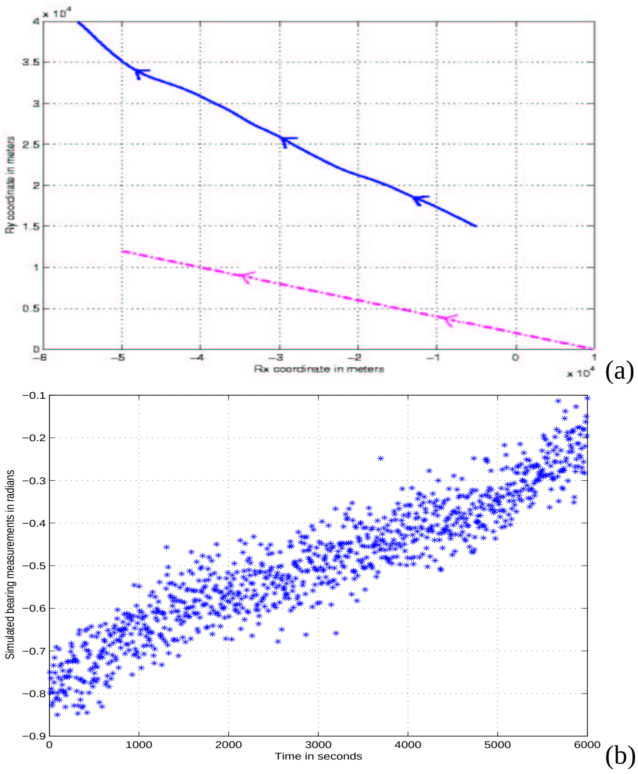

Fig. 4: *Scenario 2*: (a) trajectories of the observer (red dashed line) and the target (blue solid line). (b) Simulated bearing measurements .

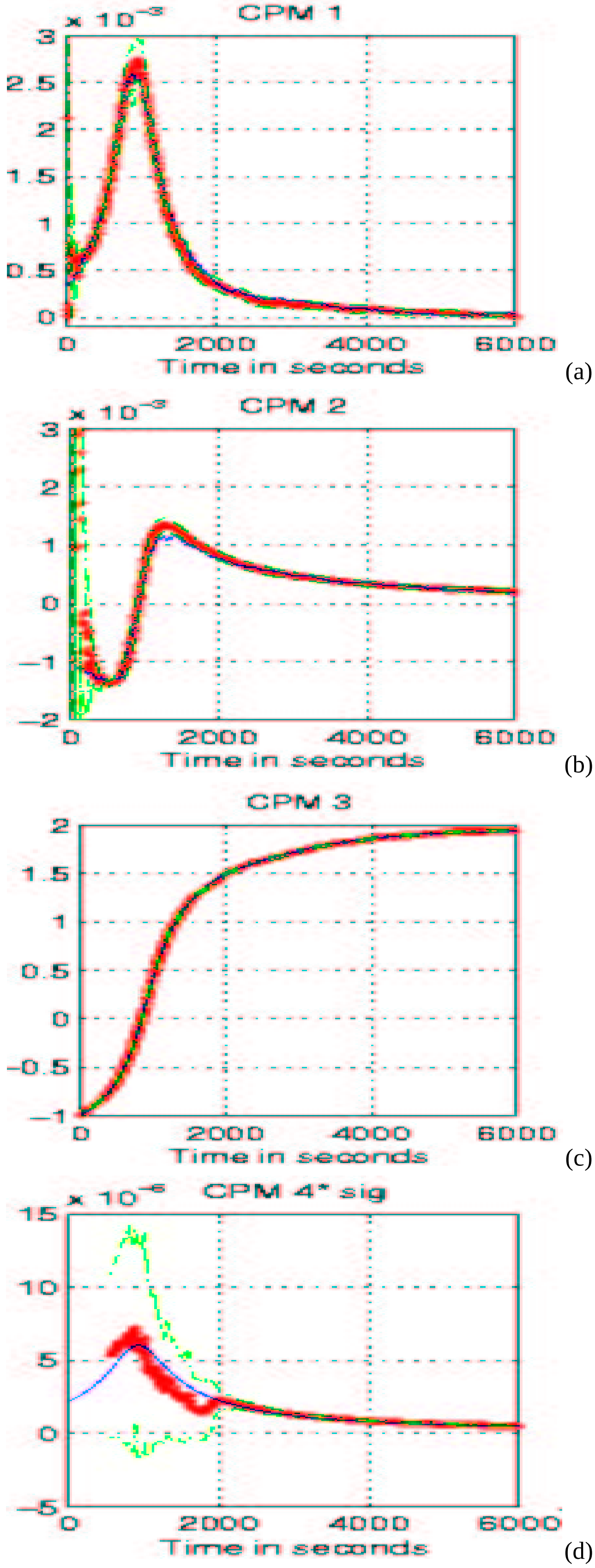

Fig. 5: Scenario 1: estimates for one particular run (red cross line), 2σ confidence bounds area in green (dotted lines). The blue line stands for the true values. (a): $Y_1(t)$, (b): $Y_2(t)$, (c): $Y_3(t)$, (d): ρ_t.

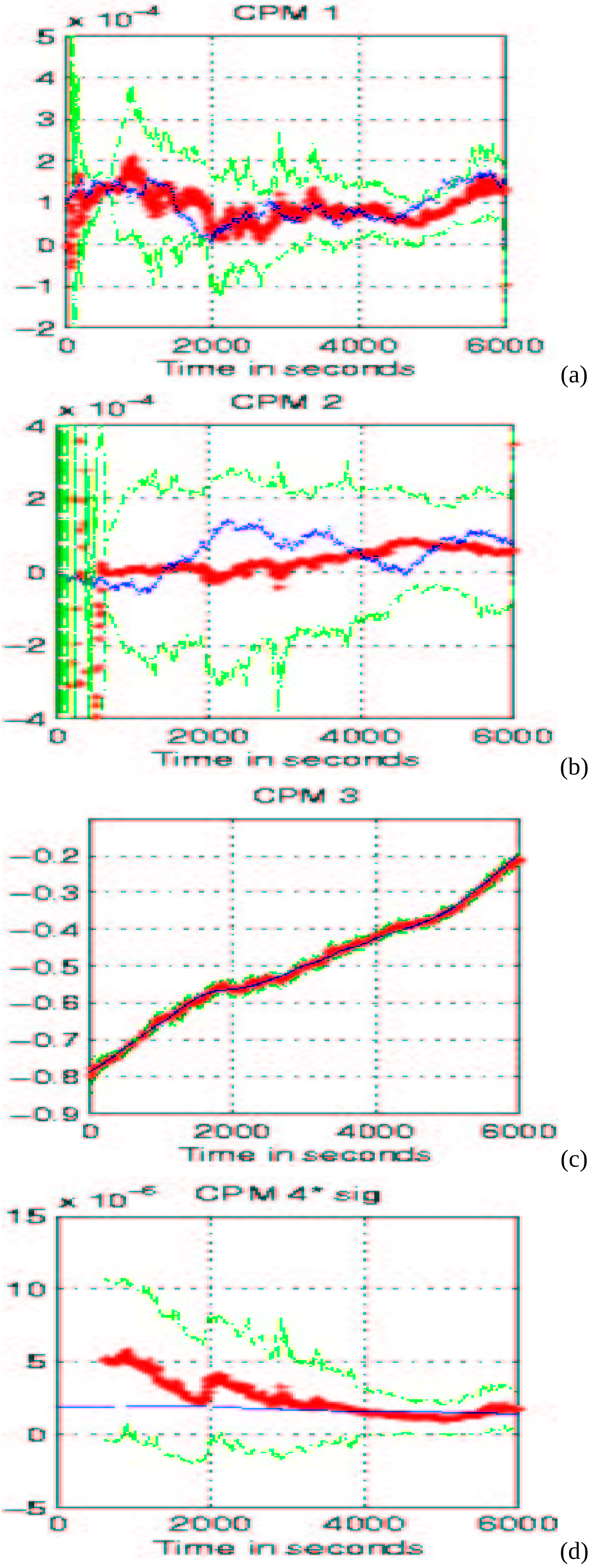

Fig. 6: Scenario 2: estimates for one particular run (red cross line), 2σ confidence bounds area in green (dotted lines). The blue line stands for the true values. (a): $Y_1(t)$, (b): $Y_2(t)$, (c): $Y_3(t)$, (d): ρ_t.